WE CLAIM:

Sub A1

1. An instruction pipe control method comprising:

reading a new instruction from an instruction pipestage,

determining, with reference to other instructions read previously from the instruction pipestage, whether valid data associated with the new instruction can be written to a next instruction pipestage and,

stalling processing of the new instruction until valid data associated with the new instruction can be written to the next instruction pipestage.

2. The method of claim 1, wherein, if the new instruction is a return instruction, the determining includes determining whether a return address is available within the instruction pipe.

3. The method of claim 1, wherein, if the new instruction is a return instruction, the determining includes determining whether sufficient time has expired from an earlier return instruction for a return address to be received from an external resource.

4. The method of claim 1, wherein, if the new instruction is a call instruction, the determining includes determining whether immediate processing of the call instruction would exceed a predetermined access rate associated with a shared resource.

5. The method of claim 4, further comprising, after the stalling terminates, storing a return address associated with the call instruction both locally and in a shared resource.

6. The method of claim 1, wherein the stalling stalls the instruction pipestage and all other instruction pipestages before it in the instruction pipe.

7. An interface method for an instruction pipe that shares access to an external resource, comprising:

reading a new instruction from an instruction pipestage,

if the new instruction requires access to the external resource, determining with reference to other instructions read previously from the instruction pipestage, whether immediate processing of the new instruction would cause the instruction pipe to exceed an access allocation for the instruction pipe, and

if so, stalling the new instruction.

8. The method of claim 7, further comprising, if the new instruction is a call instruction, after the stalling terminates, storing a return address associated with the call instruction both locally and in a shared resource.

9. The method of claim 7, wherein the stalling stalls the instruction pipestage and all other instruction pipestages before it in the instruction pipe.

10. A method for interfacing an instruction pipe with an external resource characterized by a predetermined round-trip communication latency period, the method comprising:

reading a new instruction from an instruction pipe stage,

determining, with reference to other instructions read previously from the instruction pipestage, whether valid data associated with the new instruction is available to the instruction pipe prior to expiration of the round-trip communication latency period,

if not, stalling processing of the new instruction until the round-trip communication latency period expires.

11. The method of claim 10 further comprising

determining whether the new instruction requires access to the external resource in excess of an access allocation for the instruction pipe, and

if so, stalling the new instruction.

12. The method of claim 10, wherein the stalling stalls the instruction pipestage and all other instruction pipestages before it in the instruction pipe.

13. In an instruction pipe, a clock throttling mechanism provided between a pair of instruction pipestages, comprising:

a state machine coupled to an output of a first instruction pipestage,

a clock control circuit having an input for a system clock signal and having an output for a modified clock signal, the output coupled to the pair of instruction pipestages, the clock control circuit controlled by the state machine.

14. The clock throttling mechanism of claim 13, further comprising a read/write controller under control of the state machine and having an output for controlling writes to the second instruction pipestage.

15. The clock throttling mechanism of claim 13, further comprising:

a first register coupled to the first instruction pipestage,

a second register,

a selector coupled to the first and second registers and having an output coupled to the second instruction pipestage.

16. The clock throttling mechanism of claim 15, wherein the selector is controlled by the state machine.

17. Execution logic for a processor, comprising:

a first instruction pipe, comprising a first plurality of cascaded pipestages, and

a return stack buffer provided in communication with at least one of the first pipestages; and

a second instruction pipe, comprising:

a second plurality of cascaded pipestages, at least one of the second pipestages provided in communication with the return stack buffer, and

clock throttling logic coupled to the at least one second pipestage.

18. The execution logic of claim 17, wherein the clock throttling logic comprises:

a state machine coupled to an output of the one pipestage from the second plurality,

a clock control circuit having an input for a system clock signal and having an output for a modified clock signal, the output coupled to the one pipestage, the clock control circuit controlled by the state machine.

19. The execution logic of claim 17, further comprising, in the first instruction pipe, second clock throttling logic that comprises:

a state machine coupled to an output of the one pipestage from the first plurality,

a clock control circuit having an input for a system clock signal and having an output for a modified clock signal, the output coupled to the one pipestage, the clock control circuit controlled by the state machine.

20. The execution logic of claim 17, wherein additional instruction pipestages from either the first or the second instruction pipe are provided in communication with the return stack buffer, the additional instruction pipestages also provided with additional clock throttling logic.